Multidisciplinary | Rapid Review | Open Access Journal

Received December 24, 2019, accepted January 13, 2020, date of publication January 17, 2020, date of current version January 29, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2967449

CAG: Stylometric Authorship Attribution of Multi-Author Documents Using a Co-Authorship Graph

RAHEEM SARWAR[1], NORAWIT URAILERTPRASERT[1], NATTAPOL VANNABOOT[1], CHENYUN YU[3], THANAWIN RAKTHANMANON[1,2], EKAPOL CHUANGSUWANICH[4], AND SARANA NUTANONG[1]

[1]School of Information Science and Technology, Vidyasirimedhi Institute of Science and Technology, Rayong 21210, Thailand
[2]Department of Computer Engineering, Kasetsart University, Bangkok 10900, Thailand
[3]Department of Computer Science, National University of Singapore, Singapore 119077
[4]Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok 10330, Thailand

Corresponding author: Thanawin Rakthanmanon (thanawin.r@ku.ac.th)

This work was supported in part by the Digital Economy Promotion Agency under Project MP-62-0003, and in part by the Thailand Research Fund and Office of the Higher Education Commission under Grant MRG6180266.

ABSTRACT *Stylometry* has been successfully applied to perform *authorship identification of single-author documents (AISD)*. The AISD task is concerned with identifying the *original* author of an *anonymous* document from a group of candidate authors. However, AISD techniques are not applicable to the *authorship identification of multi-author documents (AIMD)*. Unlike AISD, where each document is written by one *single* author, AIMD focuses on handling multi-author documents. Due to the combinatoric nature of documents, AIMD lacks the *ground truth* information—that is, information on writing and non-writing authors in a multi-author document—which makes this problem more challenging to solve. Previous AIMD solutions have a number of limitations: (i) the best stylometry-based AIMD solution has a low accuracy, less than 30%; (ii) increasing the number of co-authors of papers adversely affects the performance of AIMD solutions; and (iii) AIMD solutions were not designed to handle the *non-writing authors (NWAs)*. However, NWAs exist in real-world cases—that is, there are papers for which not every co-author listed has contributed as a writer. This paper proposes an AIMD framework called the Co-Authorship Graph that can be used to (i) capture the *stylistic information* of each author in a corpus of multi-author documents and (ii) make a *multi-label prediction* for a multi-author query document. We conducted extensive experimental studies on one synthetic and three real-world corpora. Experimental results show that our proposed framework (i) significantly outperformed competitive techniques; (ii) can effectively handle a larger number of co-authors in comparison with competitive techniques; and (iii) can effectively handle NWAs in multi-author documents.

INDEX TERMS Set similarity search, multi-author documents, co-authorship graph, authorship identification, stylometry, scientometrics.

I. INTRODUCTION

Stylometry has been used extensively to differentiate between the literary styles of authors [1]–[4]. Stylometry relies on the assumption that each individual author exhibits a *distinct* writing style, and it can be used to differentiate between documents written by different authors [2], [3], [3]–[7]. Stylometry has been successfully applied to solve *authorship identification* problems. Authorship identification problems aim to identify the *original* author of an *anonymous* document from a group of candidate authors [2], [8]. An *authorship identification* problem is generally solved by (i) computing the *writing style markers* (i.e., stylometric features) from documents written by candidate authors and (ii) applying a classifier to them to build a model that can identify the *original* author of an anonymous document [2]–[4], [9]–[11]. There are two main variants of the authorship identification problem. The first variant focuses on handling single-author documents (AISD), and is formally defined as follows [2].

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Liu.

Definition 1.1 (AISD): "Given a corpus $\mathcal{D}$ of *single-author* documents written by a set A of candidate authors, identify the original author of an anonymous *single-author* document from A" [2].

The second variant of authorship identification focuses on identifying the authors of a multi-author document, and is formally defined as follows [12].

Definition 1.2 (AIMD): "Given a corpus $\mathcal{D}$ of *multi-author* documents labeled with their co-authors, identify the co-authors of an anonymous *multi-author* document from a set of authors A of the given $\mathcal{D}$" [12].

The AISD problem has been extensively investigated by researchers, and most AISD techniques have shown high accuracy levels [2]–[4], [8], [9], [12], [13]. However, little attention has been paid to solving AIMD problems. In this work, we seek to provide an effective and scalable AIMD solution. Note that AISD techniques are not applicable to the AIMD problem because these techniques assume that each document is written by a *single* author. That is, AISD can be considered a *multi-class, single-label* classification problem in which each author represents a class label. In contrast, in the AIMD problem, each document is associated with multiple authors [12]. That is, AIMD can be considered a *multi-class, multi-label* classification problem (see Section II-C for details).

One of the main challenges associated with AIMD problems is that each document in the corpus is associated with multiple authors. Because of the combinatoric nature of the AIMD problem, the same group of co-authors may not be repeated in the corpus, which makes such problems more challenging to model [12]. Moreover, given a multi-author document (e.g., a scholarly article), we do not know how many authors in a co-author list contributed as writers and how many only provided ideas and feedback. That is, the AIMD problem lacks *ground truth* information, which makes it more challenging to solve. Hence, an AIMD predictive solution should be capable of attributing different fragments of the *same* collaborative/multi-author document to different authors on its co-author list without relying on absolute ground truth information [12]. To achieve such an authorship attribution capability, an AIMD solution should be capable of performing the following three main tasks [12]: (i) to capture the *stylistic information* of each individual author from a corpus of collaborative documents; (ii) to identify the non-writing author(s) from the co-author list; and (iii) to make a *multi-label* prediction for a multi-author query document [12].

Due to the rise of internet collaborative writing platforms such as GoogleDrive,[1] Wikipedia[2] and ShareLaTex,[3] the development of new techniques to handle multi-authored text is necessary. In addition, the applications of AIMD have increased in several domains such as *bibliometrics* and *information retrieval* [12]. That is, an effective and scalable AIMD framework can be used to greatly improve the processes of analyzing and measuring the *collaborative natures* of a community of researchers [12]. Instead of attributing the entire paper to all the listed authors, it can be used to perform a *fine-grained* analysis of the authorship of a scientific article. Specifically, the AIMD framework can be used to attribute *different* fragments of the *same* scientific article to different authors on the co-author list [12]. This authorship attribution capability can improve information retrieval systems in the following ways [12]. (i) Author-specific search modules can be developed for scholarly search engines to help researchers find text samples written by a specific author. (ii) Individual authors' profiles can be constructed to reflect their participation in different scientific areas. In addition to this, AIMD can be used to identify researchers who were most active in writing the articles and mentors who only provided feedback and ideas [12].

A. LIMITATIONS OF EXISTING STUDIES

1) **Constrained Scenarios.** Several AIMD studies have reported success using corpora containing collaborative scientific publications [14], [15]. However, these studies were conducted on constrained scenarios. For example, these studies identify the authors of publications based on *self-citation* information [14], [15]. Formulating citation-based solutions is limited because they are not applicable to corpora without citation information. Payer *et al.* [16] used authors' *research interest information* along with common stylometric features to perform the AIMD task. The main problem with identifying authors based on their research interests is that multiple authors/researchers may write articles on the same topic. In addition, an author's research interests may change over time.

   To handle the above issues associated with AIMD studies, we propose an AIMD framework that uses the *stylistic information of authors* only.
2) **Low Accuracy.** The performance of previous AIMD solutions needs to be greatly improved. For example, the best existing stylometry-based AIMD method [9] has less than 30% accuracy on a corpus written by more than 360 authors.
3) **Number of Co-Authors.** Existing AIMD solutions drastically decrease in accuracy as the number of co-authors on a paper increases. The best stylometry-based AIMD solution (AICD) [9] drops in accuracy from 25% to 16% as the number of co-authors increases from 2 to 7 [9].
4) **Non-writing Authors.** Previous AIMD solutions were not designed to handle *non-writing authors* [9], [16]. However, non-writing authors exist in real-world cases, as not every author listed on a paper has necessarily contributed as a *writer*.

[1] https://drive.google.com
[2] https://en.wikipedia.org/wiki/Main_Page
[3] https://www.sharelatex.com/

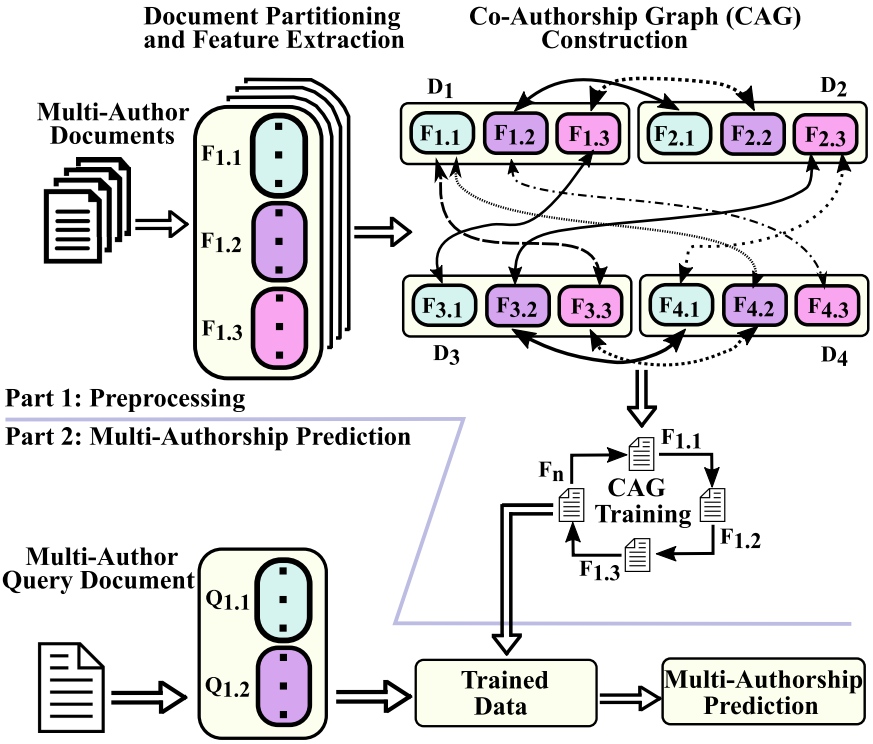


FIGURE 1. Framework Overview: Each vertex represents a fragment, and each edge between two vertices indicates that these vertices are stylistically similar. The dashed and dotted edge patterns are used to only help the distinguish overlapping crossing edges. There are two main parts of the framework: (i) preprocessing; and (ii) multi-authorship prediction. The preprocessing part of our framework is responsible for three main processes: features extraction, co-authorship graph construction (CAG) and CAG training. Once the preprocessing part was completed, we used the trained data to produce a multi-author prediction for any given query document.

B. PRELIMINARY CONFERENCE VERSION

This work is a significant extension of our preliminary conference version [12]: "A Scalable Framework for Stylometric Analysis of Multi-author Documents." We summarize our previously proposed framework with the help of Figure 1 as follows.

Our previously proposed framework consisted of two main parts: *preprocessing* and *multi-authorship prediction*. The *preprocessing* part of our framework was responsible for three main processes: *features extraction*, *co-authorship graph construction (CAG)* and *CAG training*. For the feature extraction process, we represented each multi-author document as a collection of point sets (i.e., collection of fragments). We calculated each data point from a 1000 tokens[4] using 56 stylometric features illustrated in Section Appendix VI.

After completing the feature extraction process, we constructed the CAG such that each vertex in the CAG represented a fragment, and an edge between two vertices denoted that they were *stylistically similar*. After completing CAG construction, we trained the CAG so that each fragment reflects its true author(s) only.

[4]sequences of characters separated by white spaces.

Once the preprocessing part of our solution was completed, we used the *trained data* to produce a multi-author prediction for any given query document. To derive a probabilistic prediction for a given query sample, we used the probabilistic labels of stylistically similar samples in the query sample. By doing so, we were able to effectively capture the document's collaborative nature in both the test and the training samples. Each part is described in more detail in Section III.

C. PROPOSED WORK AND OUR CONTRIBUTIONS

In this work, we improve the following aspects of our previously proposed framework [12].

- **Accuracy Improvement:** Although our proposed framework in the preliminary conference version of this work significantly outperformed the state-of-the-art stylometry-based AIMD solution, it is still necessary to greatly improve the performance of AIMD. The preliminary version of our framework has an accuracy level of 76.92% using a corpus containing 3,600 multi-author documents written by 1,360 authors, where each document was written by 3 authors and the number of non-writing authors (NWAs) was set to 2. To achieve this objective, we use the *character n-grams* as the set of features. The usual approach to using character *n*-grams

as a set of features is to choose 5-grams, 4-grams or 3-grams [17]–[23], or use the variable length word or character n-grams [24]–[26]. In this work, we use *variable length* character *n*-grams to handle multi-author documents by incorporating them into our previous feature space (see Section III-A for more details).
Character *n*-grams are a contiguous sequence of *n* characters from a text sample. For example, the character 3-grams of the beginning of this sentence would be "For," "or␣," "r␣ e," "␣ ex," etc. The motivations for incorporating character *n*-gram-based features into our existing feature space are five-fold [17]–[23]: (i) Character *n*-gram-based features have been proven to perform well in solving authorship identification problems regardless of the length of text samples. Specifically, the character *n*-gram features can effectively capture the *stylistic information* of the authors from smaller text samples (i.e., 500 tokens) compared to vocabulary based features. The character *n*-gram features provide the best results when the value of *n* is 5, 4 or 3 [18]–[23]. (ii) Character *n*-gram features can capture *complicated stylistic* information about authors on the syntactic, structural and lexical levels [18]. (iii) Character *n*-grams can tolerate *noise* in text samples (i.e., "stilometric" and "stylometric" have many common character 3-grams) [17]. (iv) Character *n*-grams require high-dimensional representation, which is not easy for humans to understand. Thus, attempts at deception are likely to fail [17]. (v) Extracting character *n*-grams does not require tokenizers, taggers, parsers or any language-dependent and non-trivial NLP tools, which makes them feasible for performing *multi-lingual* authorship attribution tasks.

- **Handling Short Documents:** Our proposed framework in the preliminary conference version of this work is capable of handling documents of 12,000 tokens or more. Our previous feature space required 1,000 tokens to obtain reliable stylometric information from each chunk (data point) [12]. In this work, we aim to improve our previously proposed framework such that it can also handle short documents, i.e., documents of 6,000 tokens or more, e.g., short scholarly publications (see Section III-A for more details).
- **Multilingual AIMD:** Most AIMD studies focused on English corpora, mainly because after World War II, English became the *lingua franca* throughout much of the world. However, bibliometric databases contain significant numbers of non-English scholarly publications. For example, Amano *et al.* [27] reported that searching for scientific literature on Google Scholar using keywords such as "biodiversity" and "conservation" generated 75,513 publications, of which 35.6% were written in non-English languages. There is clearly a substantial need to formulate multilingual AIMD solutions. The main difference between monolingual and multilingual NLI techniques is that the former is designed for a specific language. This technique may not be applicable to other languages because of linguistic differences. In contrast, a multilingual AIMD framework is a generalized solution that can be applied to different languages. That is, a multilingual AIMD framework must be able to achieve similar accuracy rates across different languages. To achieve this objective, we formulate a *multilingual* feature space which makes our solution applicable to different languages. Specifically, our feature space for multilingual AIMD relies on a minimal linguistic assumption set that includes (i) the ability to tokenize a text sample into words, (ii) the ability to identify sentence boundaries, (iii) the capability of POS tagging and (iv) the use of punctuation. We perform experiments on four languages: English, French, Finnish and German (see Section IV-D.1 for more details).

1) RESEARCH QUESTIONS

In addition to addressing the aforementioned limitations of existing studies and the improvements in the AIMD framework, we answer the following research questions in this paper.

1) **Research Question 1.** Recall that, we use the *character n-grams* as the set of features and incorporate them into our previous feature space. Thus, we investigate how important it is to use character n-grams features for authorship identification of *multi-author* documents along with the other stylometric features such as lexical, syntactic and structural features?
2) **Research Question 2.** Recall that, in order to make our solution robust and generalizable to other languages, we formulate a feature space that relies on a minimal linguistic assumption set. How robust is our feature space when applied to multilingual settings?

To achieve the objective of this study, we improve our framework as follows.

2) SUMMARY OF CONTRIBUTIONS IN THIS WORK

- We use the *character n-grams* as the set of features. The usual approach to using character *n*-grams as a set of features is to choose 5-grams, 4-grams or 3-grams [17]–[23], or use the variable length word or character n-grams [24]–[26]. In this work, we use *variable length* character *n*-grams to handle multi-author documents by incorporating them into our previous feature space.
- We formulate a *multilingual* feature space which makes our solution applicable to different languages. We perform experiments on four languages: English, French, Finnish and German.
- We create new synthetic corpora for the AIMD task which is 100% larger than the existing synthetic corpus

in terms of authors and the number of multi-author documents.
- Our proposed technique in the preliminary conference version of this work was tested on two real-world corpora. In this work, we create another new real-world corpus and we test our proposed solution on three different real-world corpora.
- We extensively compare the performance of our solution against the baseline method; the best existing AIMD solution and its improved variation; and well-known multi-label classification methods.

For rest of the paper, section 2 reviews the existing studies on the authorship identification problem and its variations. Section 3 elaborates our proposed framework. Section 4 reports the findings of this investigation. Section 5 presents concluding remarks and future research directions.

II. LITERATURE REVIEW

A. STYLOMETRY

Stylometry has been used extensively to differentiate between the literary styles of authors [2]–[4] Stylometry relies on the assumption that each individual author exhibits a *distinct* writing style, and it can be used to differentiate between documents written by different authors [2]–[4]. and has been used to solve authorship-related problems such as authorship verification, author profiling and authorship identification [2]–[4], [13], [28]–[31].

1) STYLOMETRIC FEATURES

Stylometric features are writing style markers that can be used to differentiate between documents written by different authors. Studies have proposed various stylometric features, including lexical, syntactic, structural and idiosyncratic features [9], [16], [32]–[34].

- Lexical features can be defined as statistical measures of word-based and character-based lexical variations in the text, such as vocabulary richness [32], word length distributions and character *n*-gram-based features [33].
- Examples of syntactic features include *function words* and *part-of-speech* tags [34].
- The structural features are writing style markers based on the presentation of the text, such as the average number of words in a paragraph or in a sentence [33].
- The idiosyncratic features are associated with the errors in the text samples of an author, such as grammatical mistakes and misspellings [35].

Payer *et al.* [16] used topic information and common stylometric features to perform the AIMD task. They used a set of 10,727 features, of which 7,954 were citation-based, 2,374 were content-based and only 399 were stylometric features. Later, Dauber *et al.* [9] proposed an AIMD technique using the *Writeprints Limited* feature set [36]. This set includes five types of features: (i) idiosyncratic, (ii) content-specific, (iii) structural, (iv) lexical and (iv) syntactic features.

2) COMPARISON TO OUR WORK

In this work, we used a set of 1,056 writing style markers (stylometric features). These features can be organized into three categories: lexical, structural and syntactic [32]–[34]. Our feature set is described in Section Appendix VI.

One main difference between our solution and the majority of previous solutions is that ours uses authors' stylistic information to perform AIMD. The main advantages of formulating a stylometry-based AIMD solution over the majority of previous AIMD solutions are twofold: (i) unlike most previous AIMD studies [14]–[16], our AIMD solution is applicable to the *multi-author corpora* without the citation information, and (ii) our AIMD solution is capable of handling different authors working on the same topic and authors whose research interests change over time. In addition, our feature space contains a set of 1,056 features, which is smaller than the feature sets used in previous AIMD studies [14]–[16]. Thus, our feature space is computationally less expensive and requires less storage than the feature spaces used in past AIMD studies.

B. AIMD

The authorship identification task has two main variations. The first focuses on handling single-author documents (AISD) [2], while the second focuses on handling multi-author documents (AIMD) [12], and is formally defined as follows. Given a dataset $\mathcal{D}$ of *multi-author* documents labeled with their *co-authors*, identify the co-authors of an anonymous *multi-author* document from a set of authors A in a given $\mathcal{D}$ [12]. There are several variations of the AIMD problem. These AIMD variations are comparativly easier to solve and have shown promising performance. For example, one AIMD variation uses *single-author* documents for model training, which makes it easy to solve, as training a model using a set of multi-author documents is challenging because of the lack of ground truth information [12]. When each document is associated with multiple authors, we do not how many of them contributed as writers and how many only provided ideas and feedback. However, this AIMD variation may not be viable when the training samples are also multi-author documents [9], [37] (i.e., most scientific papers have more than one author). Moreover, this variation in AIMD [9] shows a substantial drop in performance when the number of co-authors increases. For example, increasing the number of co-authors from two to three in a test document reduces the accuracy from 50% to 30% [9]. Another AIMD variation assumes that each group of co-authors has a sufficient number of documents for model training [9]. However, because of the *combinatoric* nature of collaborative patterns in a community of researchers, we consider this assumption to be *unrealistic* [12].

To evaluate the performance of our proposed framework (*Co-Authorship Graph (CAG)*), we compare it against: (i) the best existing stylometry-based AIMD technique, AICD [9], (ii) the improved version of AICD, (I-AICD) and (iii) the

baseline technique proposed in the preliminary conference version of this investigation (B-CAG).

On the other hand, the algorithm adaption techniques extend and adopts a specific classification method to handle multi-label classification problem, such as, multi-label k-nearest neighbor (MLkNN), and multi-label decision trees (ML-DT). To evaluate the performance of our proposed framework (*Co-Authorship Graph (CAG)*), we compare it against the aforementioned multi-label learning methods. Descriptions of all of these competitive methods are given in section IV-A.

As mentioned in section Introduction that AIMD can be considered a *multi-class, multi-label* classification problem. The following subsection reviews multi-label learning techniques.

C. MULTI-LABEL CLASSIFICATION TECHNIQUES AND EXISTING AIMD TECHNIQUES

For single-label classification tasks, each instance in the training data is associated with one class label "l" from a disjoint label set L [38], [39]. When $|L| = 2$, a learning problem is known as binary classification problem (e.g., the gender identification problem, where the task is to assign an anonymous text to one of two classes, i.e., male or female) [40], [41]. When $|L| > 2$, a learning problem is known as a multi-class classification problem (e.g., authorship identification of single-author documents) [12].

For multi-label classification tasks, each sample in the training dataset is associated with a set of class labels, and the task is to predict the label set of an unseen test sample. For example, in the *scene classification problem*, each image may be associated with many semantic classes, e.g., beach and urban [38]. In the *functional genomics classification problem*, where each gene may belong to a set of functional classes, such as transcription, protein synthesis and metabolism [42]. In *authorship identification of multi-author documents*, each document is associated with multiple authors [16]. In these examples, each sample in the training dataset is associated with a set of class labels, and the task is to predict the label set of an unseen test sample. It is clear that this multi-label classification problem fits the AIMD problem definition [12]. The rest of this subsection reviews multi-label learning techniques.

Previous multi-label learning techniques can be organized into two categories: (i) problem transformation techniques, and (ii) algorithm adaptation techniques. Problem transformation techniques transform a multi-label classification problem into a *multi-class, single label* classification problem. By doing so, a *multi-label* classification problem can be solved using a *single-label* learning method. The examples of such methods includes *copy transformation* (CT) [43], *binary relevance* (BR) [38], *ensemble classifier chain* (ECC) [44], *directed acyclic graph* (DAG) based method [45], and a recently proposed method presented in the paper entitled *leveraging label-specific discriminant mapping features* (LSDM) [46].

On the other hand, the algorithm adaption techniques extend and adopts a specific classification method to handle multi-label classification problem, such as, multi-label k-nearest neighbor (MLkNN), and multi-label decision trees (ML-DT). To evaluate the performance of our proposed framework (*Co-Authorship Graph (CAG)*), we compare it against the aforementioned multi-label learning methods. Descriptions of all of these competitive methods are given in section IV-A.

D. SET SIMILARITY SEARCH

In this subsection, we briefly review the set similarity measures because our proposed AIMD framework relies on the identification of stylistically similar document fragments, where each fragment is represented as a point set in a real-valued vector space. The *standard Hausdorff distance (SHD)* is a well-known set distance measure that can be used to compute the distance between two point sets in a real-valued vector space. SHD is defined as

$$H(Q, F) = \max\{h(Q, F), h(F, Q)\}, \tag{1}$$

where

$$h(Q, F) = \max_{q_i \in Q} \min_{f_j \in F} d(q_i - f_j) \tag{2}$$

and $d(.,.)$ refers to a distance function for comparing the data points q_i and f_j. The function $h(Q, F)$ is the directed Hausdorff distance from set Q to set F. The function $h(Q, F)$ identifies the data point q in Q that is farthest from any data point of set F and calculates the distance from q to its nearest neighbor in the set F using the distance function d. Thus, the SHD distance, $H(Q, F)$, is used to calculate the degree of mismatch between two point sets because it identifies the distance of the point from set Q that is the farthest of any data point from F, and vice versa. In other words, the two point sets Q and F are considered similar *iff* for every point of set Q, there is at least one point in set F in proximity, and vice versa [2].

We note that $h(Q, F)$ cannot be considered a metric distance function because it does not satisfy the *identity of indiscernibles* principle and the symmetry property does not hold for it. For brevity, we use the term *distance* to refer to $h(Q, F)$.

Researchers have argued that SHD is sensitive to outliers. That is, a single outlier data point significantly affects the distance value [47], [48]. To mitigate the outlier sensitivity issue associated with SHD, researchers formulated two variants of SHD: *modified Hausdorff distance (MHD)* [47] and *partial Hausdorff distance (PHD)* [48]. The MHD can be computed by (i) ranking all data points in Q according to the minimum distance to F; and (ii) computing the average of the minimum distances within a given range, i.e., (50%, 100%] [47]. The difference between MHD and PHD is that, the second parameter is always 100% for MHD. We test these three distance measures and find that MHD performs better

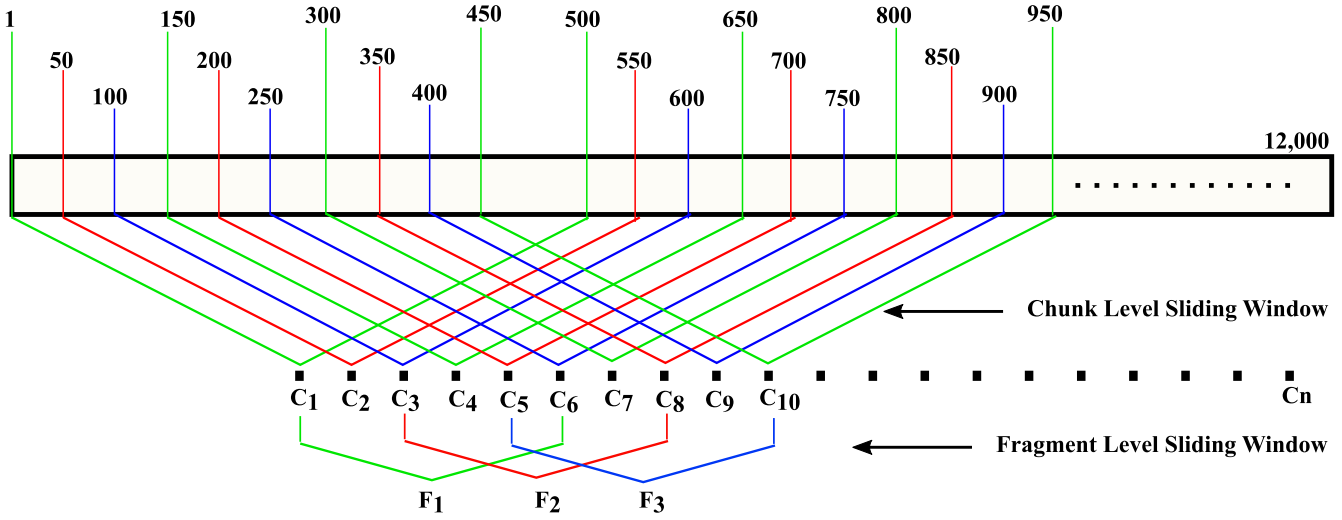


FIGURE 2. The Feature Extraction Process Using Sliding Window: Using 500 tokens as the size of sliding window and 50 tokens as the sliding window increment, we can produce 231 data points for each 12,000-token document. We apply the same principle at the fragment level to obtain a sufficient number of fragments to conduct our analysis.

than PHD and MHD. We thus provide experimental results based on MHD only (see Section III-C for more details).

E. SUMMARY

This subsection summarizes the main differences between our framework and the majority of other AIMD studies.

- Unlike other AIMD studies, we represent each document in the corpus as a collection of point sets (see Section III-A for more details).
- Unlike any other AIMD technique, our graph training algorithm is effective at (i) learning the true writer(s) of each fragment and (ii) identifying the NWAs of multi-author documents (see Section III-C for more details).
- Several previous AIMD techniques are not applicable to corpora without citation information or are unable to handle authors whose research interests change over time. In contrast, our framework uses the stylistic information of authors in performing AIMD. Thus, unlike most previous AIMD studies, our solution is applicable to corpora without citation information, to different authors working on the same topic and to authors whose research interests change over time.
- Our solution is applicable to multiple languages.

III. PROPOSED SOLUTION

In this section, we discuss the proposed framework, which consists of two main parts: *preprocessing* and *multi-authorship prediction*. The *preprocessing* part of our solution is responsible for three main processes: (I) *feature extraction*, (II) *co-authorship graph construction*, and (III) *co-authorship* graph training. Once the preprocessing part is finished, we use the trained data to produce a multi-author prediction for any given query document. The following subsections explain each part of our proposed framework in detail.

A. PREPROCESSING: FEATURE EXTRACTION

The *preprocessing* part of our framework is responsible for three main processes: (i) *features extraction*, (ii) *co-authorship graph (CAG) construction* and (iii) *CAG training*. In this subsection, we provide a detailed discussion of the feature extraction process.

As explained earlier, we represent each document in the corpus as a collection of fragments (i.e., a collection of point sets) where each fragment is represented as a point set. We calculate each data point from a 500 tokens[5] using the features illustrated in Section Appendix VI. There are two main motivations for representing each document as a *collection of point sets*. (i) It allows us to attribute different fragments of the same multi-author document to their original authors in the author list. (ii) It allows us to apply *set similarity measures* associated with outlier handling techniques, such as the *modified Hausdorff distance* [47], which can help improve the performance of our framework.

To obtain reliable stylistic information from each data point, we set the size of each data point to 500 tokens. However, using 500 tokens per chunk results in only 24 data points for a 12,000-tokens document, which are insufficient for our stylometric analysis. To overcome this issue, we apply the concept of the *sliding window* to generate data points with overlapping token sequences. We explain this process in Figure 2. For example, using 500 tokens as the size of sliding window and 50 tokens as the sliding window increment, we can produce 231 data points for each 12,000-token document. We apply the same principle at the fragment level to obtain a sufficient number of fragments to conduct our analysis. Specifically, by setting the fragment size to 6 data points and the sliding window increment value to 2, we can generate 113 fragments for each 12,000-token document.

[5]sequences of characters separated by white spaces.

Algorithm 1 Co-Authorship Graph (CAG) Construction

```
1:  procedure CAGConstruction
2:      Vertices ← []
3:      Edges ← []
4:      for F in Fragments do
5:          Neighbors ← GetKNN(F, Fragments)
6:          for N in Neighbors do
7:              Edges.Append(F, N)
8:          end for
9:          F.PMF ← GenerateUniformPMF(F.AuthorList)
10:         Vertices.Append(F)
11:     end for
12:     return G(Vertices, Edges)
13: end procedure
```

As mentioned earlier in Introduction that, in this work, we use *variable length* character *n*-grams to handle multi-author documents by incorporating them into our previous feature space. Specifically, we extract 3-grams, 4-grams and 5-grams from a corpus of multi-author documents. Once we measure their *term frequency-inverse document frequency (tf-idf)* scores, we rank them in descending order according to their *tf-idf* scores to select the top 1,000 *n*-grams to use as features along with our previous feature space.

Character *n*-grams are a contiguous sequence of *n* characters from a text sample. For example, the character 3-grams of the beginning of this sentence would be "For," "or␣," "r␣ e," "␣ ex," etc. The motivations for incorporating character *n*-gram-based features into our existing feature space are five-fold [17]–[23]: (i) Character *n*-gram-based features have been proven to perform well in solving authorship identification problems regardless of the length of text samples. Specifically, the character *n*-gram features can effectively capture the *stylistic information* of the authors from smaller text samples (i.e., 500 tokens) compared to vocabulary based features. The character *n*-gram features provide the best results when the value of *n* is 5, 4 or 3 [18]–[23]. (ii) Character *n*-gram features can capture *complicated stylistic* information about authors on the syntactic, structural and lexical levels [18]. (iii) Character *n*-grams can tolerate *noise* in text samples (i.e., "stilometric" and "stylometric" have many common character 3-grams) [17]. (iv) Character *n*-grams require high-dimensional representation, which is not easy for humans to understand. Thus, attempts at deception are likely to fail [17]. (v) Extracting character *n*-grams does not require tokenizers, taggers, parsers or any language-dependent and non-trivial NLP tools, which makes them feasible for performing *multi-lingual* authorship attribution tasks.

B. PREPROCESSING: CAG CONSTRUCTION

Recall that one of the main challenges associated with the AIMD problem is that each document is associated with multiple authors. Because of its combinatoric nature, the same author list may not be repeated in a corpus of multi-author documents, which makes this problem more challenging to model. Moreover, in a multi-author document (e.g., a scientific article), some of the authors on the author list may not have contributed as writers [12]. That is, AIMD problems lack ground truth information, which makes this problem more challenging to solve. An AIMD predictive method should therefore be capable of inferring the authorships of a multi-author document without absolute ground truth information. Our proposed AIMD framework relies on the observation that *the stylistically similar fragments are likely to have been written by a similar group of authors*. To capture the *stylistic similarities* of document fragments we propose a data structure called the *Co-authorship Graph (CAG)*. In addition, we propose an iterative algorithm to identify the original author of each document fragment.

After completing the feature extraction process, we construct a co-authorship graph (CAG) in which each vertex represents a fragment and an edge between two vertices shows that they are stylistically similar. Algorithm 1 provides the structure of the CAG construction process. Recall that after the completion of the feature extraction process, each document in the corpus is represented as a collection of point sets (collection of fragments), where each data point corresponds to a feature vector. As shown in Algorithm 1, we iterate through all of the fragments from all of the documents in the training corpus (Lines 4 to 10). To construct the CAG edges, we identify k stylistically similar fragments for each fragment, where the *modified Hausdorff distance (MHD)* [47] is used as the distance between two fragments. Specifically, the GetKNN(F, Fragments) procedure identifies the k fragments in *Fragments* with the smallest MHDs with respect to F (Line 5). These nearest neighbors are the vertices of the graph, and the MHD distances are the weights of the edges. We assume that each fragment F of a document is associated with a list of authors F.*AuthorList* from the document, which might include one or more NWAs [12]. We initialized the authors by assigning the same probability to each author in the author list (Lines 9 to 10). After iterating through *all of the fragments from all of the documents*, the CAG is returned (Line 12).

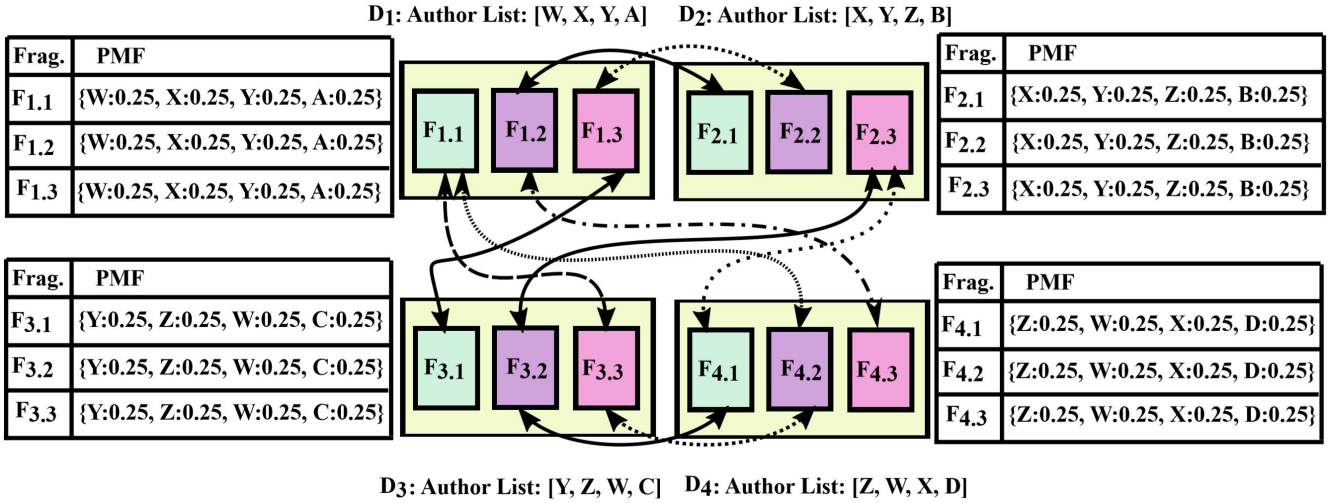


FIGURE 3. CAG: Each vertex represents a fragment, and each edge between two vertices indicates that these vertices are stylistically similar. The dashed and dotted edge patterns are used to only help the distinguish overlapping crossing edges. The adjacent tables of all of the fragments show the initial PMFs.

Algorithm 2 CAG Training

```
procedure UpdateCAGVertex
    NeighborPMFs ← []
    NeighborDistances ← []
    V ← ThisVertex
    for N in V.GetNeighbors() do
        PMF ← ReceivePMF(N)
        PMF ← RemoveNWAs(PMF, V.AuthorList)
        PMF ← Renormalize(PMF)
        NeighborPMFs.Append(PMF)
        NeighborDistances.Append(Distance(V, N))
    end for
    V.PMF ← ComputeWeightedAvg(NeighborPMFs, NeighborDistances)
    for N in V.GetNeighbors() do
        SendPMF(N, V.PMF)
    end for
end procedure
```

C. PREPROCESSING: CAG TRAINING

Now we explain the CAG training process using the example shown in Figure 3. There are four multi-author documents, where each document is associated with four authors. We set the ground truth information as follows. First, we assume that only the first three authors of each document contributed as writers. For example, for the document D_1, only Authors W, X and Y wrote parts of D_1, while author A was a non-writing author (NWA). Similarly, Authors B, C and D are NWAs of D_2, D_3 and D_4, respectively. We note that this ground truth information is hidden from the model.

We also illustrate the initial PMF of each document fragment in Figure 3. As *the ground truth information about the non-writing authors is hidden from the model*, we associate each fragment of a document with all the listed co-authors with an equal probability distributed among them. For example, the author PMFs of $F_{1.1}$, $F_{1.2}$, $F_{1.3}$ are uniform, e.g., $\{W : 0.25, X : 0.25, Y : 0.25, Z : 0.25\}$. We derive the initial PMFs of rest of all fragments shown in the figure in the same fashion.

The CAG construction algorithm (Algorithm 1) returns edges that connect stylistically similar fragments. For example, based on the identified edges, it can be seen that fragment $F_{1.1}$ is stylistically similar to $F_{3.3}$ and $F_{4.2}$. Similarly, $F_{1.2}$ is stylistically similar to $F_{2.1}$ and $F_{4.3}$. Despite the fact that each fragment in a multi-author document is initialized with an equal probability distributed among all co-authors, they are connected with different sets of stylistically similar fragments that are associated with different co-authors.

Next, we show that how these differences in the author lists of fragments can collaboratively identify authors who contributed as writers using Algorithm 2.

Algorithm 3 Authorship Identification

```
1:  procedure Multi-AuthorshipPrediction
2:      FragmentPMFs ← []
3:      QueryFragments ← GetDocumentFragments(Q)
4:      for Q in QueryFragments do
5:          Neighbors ← GetKNN(Q, Fragments)
6:          NeighborPMFs ← []
7:          for N in Neighbors do
8:              NeighborPMFs.Append(PMF)
9:              NeighborDistances.Append(Distance(Q, N))
10:         end for
11:         Q.PMF ← ComputeWeightedAvg (NeighborPMFs, NeighborDistances)
12:         FragmentPMFs.Append(Q.PMF)
13:     end for
14:     return GetDocumentPMF(FragmentPMFs)
15: end procedure
```

There are two main objective of the *Co-Authorship Graph (CAG)* training algorithm: (i) to alter the PMF of each fragment such that it reflects the true author(s) of that fragment and (ii) to remove non-writing authors from the author list. The process of updating the PMF of each vertex is given in Algorithm 2. We execute the same algorithm (Algorithm 2) at each vertex in the corpus with multiple iterations called *supersteps*. Each vertex represents a *fragment* and each edge between two nodes denotes that they are stylistically similar fragments. In this algorithm, each vertex makes note of the top$-k$ most similar fragments as neighbors. This algorithm consists of three main parts: (i) *Receive*, (ii) *Compute* and (iii) *Send*.

- *Receive PMF(Lines 5 to 10).* The vertex receives the PMFs from its top-k most similar fragments (neighbors).
- *Update PMF (Line 12).* The vertex PMF is updated as the weighted average of all neighbors' PMFs. These weights are obtained from the distances of the neighbors using the *Probabilistic k Nearest Neighbor (PkNN* method with the *radial basis function (Gaussian) kernel* [49]. The total weight is assumed to be normalized to 1 [12].
- *Send Updated PMF (Lines 13 to 14).* The updated PMF is sent to the neighbors.

At each superstep, we apply the same process in Algorithm 2 and repeat the supersteps until all of the PMFs are converged or the number of iterations reaches a specified value, which is 15 in this study. Consider now how Algorithm 2 operates in the context of the example given in Figure 3. The fragment $F_{1.1}$ receives two PMFs from its two neighbors $F_{3.3}$ and $F_{4.2}$ as $\{Y : 0.25, Z : 0.25, W : 0.25, C : 0.25\}$ and $\{Z : 0.25, W : 0.25, X : 0.25, D : 0.25\}$, respectively (Line 6). We then compare the PMF of each fragment (neighbor) against the co-author list $[W, X, Y, A]$ to remove author(s) who do not appear in the co-author list of $F_{1.1}$ (Line 7). In this example, the authors Z and C are discarded from the fragment $F_{3.3}$. Similarly, authors Z and D are discarded from the fragment $F_{4.2}$. After discarding the authors who do not appear in the co-author list of $F_{1.1}$, we re-normalize the PMF. The re-normalization results in $\{Y : 0.5, W : 0.5\}$ as the PMF for $F_{3.3}$ and $\{W : 0.5, X : 0.5\}$ as the PMFs for $F_{4.2}$. For ease of exposition, we assume that the two nearest neighbors are the same distance from the respective fragment and contribute equally to the fragment's PMF. Hence, the weighted average of the two PMFs is $\{W : 0.5, X : 0.25, Y : 0.25\}$ after the first superstep.

Following the same process, we obtain

- $\{W : 0.25, X : 0.5, Y : 0.25\}$ for $F_{1.2}$,
- $\{W : 0.25, X : 0.25, Y : 0.5\}$ for $F_{1.3}$,
- $\{X : 0.5, Y : 0.25, Z : 0.25\}$ for $F_{2.1}$,
- $\{X : 0.25, Y : 0.5, Z : 0.25\}$ for $F_{2.2}$,
- $\{X : 0.25, Y : 0.25, Z : 0.5\}$ for $F_{2.3}$,
- $\{Y : 0.5, Z : 0.25, W : 0.25\}$ for $F_{3.1}$,
- $\{Y : 0.25, Z : 0.5, W : 0.25\}$ for $F_{3.2}$,
- $\{Y : 0.25, Z : 0.25, W : 0.5\}$ for $F_{3.3}$,
- $\{Z : 0.5, W : 0.25, X : 0.25\}$ for $F_{4.1}$,
- $\{Z : 0.25, W : 0.5, X : 0.25\}$ for $F_{4.2}$, and
- $\{Z : 0.25, W : 0.25, X : 0.5\}$ for $F_{4.3}$.

As can be seen, all of the PMFs become less uniform after only the first superstep. For each document, the PMFs converge to the following values [12].

1) *Document* D_1:
   $\{W : 1\}$ for $F_{1.1}$, $\{X : 1\}$ for $F_{1.2}$, and $\{Y : 1\}$ for $F_{1.3}$.
2) *Document* D_2:
   $\{X : 1\}$ for $F_{2.1}$, $\{Y : 1\}$ for $F_{2.2}$, and $\{Z : 1\}$ for $F_{2.3}$.
3) *Document* D_3:
   $\{Y : 1\}$ for $F_{3.1}$, $\{Z : 1\}$ for $F_{3.2}$, and $\{W : 1\}$ for $F_{3.3}$.
4) *Document* D_4:
   $\{Z : 1\}$ for $F_{4.1}$, $\{W : 1\}$ for $F_{4.2}$, and $\{X : 1\}$ for $F_{4.3}$.

The NWAs of each document are not included in the PMFs, and the author lists of D_1, D_2 and D_3 are correctly identified as $[W, X, Y]$, $[X, Y, Z]$, $[Y, Z, W]$ and $[Z, W, X]$, respectively.

D. MULTI-AUTHORSHIP PREDICTION

Given a multi-author query document $\mathcal{Q}$, our framework makes a multi-authorship prediction using training fragments from the co-authorship graph training step (cf. Algorithm 2). The structure of the multi-authorship prediction process is given in Algorithm 3. As can be seen in Algorithm 3, we decompose $\mathcal{Q}$ into a set of query fragments (Line 3). For each query fragment Q (Lines 5 to 10), the top-k nearest neighbors are identified using the GetKNN() function introduced in Algorithm 1 for CAG construction. Similar to the CAG training process given in Algorithm 2, we compute the weighted average to make a single prediction for each query fragment Q by using the PMFs of the neighboring fragments and their distances with respect to Q. After obtaining the PMFs of all of the query fragments (Line 12), the next step is to combine the PMFs of all of the query fragments to make a final prediction for the entire $\mathcal{Q}$. Specifically, we compute the average PMF to make a final prediction for the entire $\mathcal{Q}$.

We realize this concept with the help of Figure 3. Assume that a query document $\mathcal{Q}$ is decomposed into two query fragments $Q_{1.1}$ and $Q_{1.2}$. We also assume that $F_{1.1}$ and $F_{3.3}$ were identified as the two nearest neighbors of $Q_{1.1}$ and that $F_{1.3}$ and $F_{3.1}$ are identified as the two nearest neighbors of $Q_{1.2}$. We can then obtain the predictions of these two query fragments (i.e., $Q_{1.1}$ and $Q_{1.2}$) as the PMFs $\{W : 1.0\}$ and $\{Y : 1.0\}$, respectively. The final prediction of the entire query document $\mathcal{Q}$ is $\{W : 0.5, Y : 0.5\}$; i.e., W and Y are the authors of the query document $\mathcal{Q}$.

IV. PERFORMANCE EVALUATION

This section describes the *competitive methods*, provides description of synthetic and real-world corpora, illustrates the *experimental setup* and reports the findings obtained from extensive experimental studies.

A. COMPETITIVE METHODS

To evaluate the performance of our proposed framework (*Co-Authorship Graph (CAG)*), we compare its performance against: (i) the best existing stylometry-based AIMD technique, AICD [9], (ii) the improved version of AICD, (I-AICD) and (iii) the baseline technique proposed in the preliminary conference version of this investigation (B-CAG). In addition, we compare the performance our proposed framework against well-known multi-label classification methods using our feature space: *copy transformation* (CT) [43], *binary relevance* (BR) [38], *ensemble classifier chain* (ECC) [44], *directed acyclic graph* (DAG) based method [45], recently proposed method presented in the paper entitled *leveraging label-specific discriminant mapping features* (LSDM) [46], and multi-label k-nearest neighbor (MLkNN). Description of all of these competitive methods are as follows.

1) DESCRIPTION OF EXISTING AIMD COMPETITIVE METHODS

- **AICD.** Inthis investigation, we abbreviate the best existing stylometry-based AIMD solution to AICD. This solution is presented in the paper entitled *stylometric authorship identification of collaborative documents* [9]. AICD uses the linear *support vector machine (SVM)* classification method. For the training data, AICD relies on the *copy transformation* technique. For each multi-author training sample associated with m labels, the copy transformation technique creates m single-label samples [43], each of which can be associated with one label at a time [43]. For the feature space, AICD extracts the *WritePrints Limited feature set* [36] from a corpus of multi-author documents using the *Jstylo* tool [50]. In order to make a multi-author prediction for a multi-author query document, AICD converts the output of the classifier into a probabilistic distribution and uses the most probable m authors as a result [9].
- **I-AICD.** We formulate an improved variation of the best existing stylometry-based competitive technique (AICD), which is abbreviated to I-AICD in this investigation. Similar to our proposed solution, for I-AICD, we also use the *sliding window technique* to generate chunks from each multi-author document following the same method used for AICD. At this stage, we aggregate the prediction at the chunk level by having each chunk vote for its most likely author.
- **B-CAG.** We also compare our solution against the baseline technique proposed in the preliminary version of this work called *Baseline Co-Authorship Graph (B-CAG)* [12].

2) DESCRIPTION OF MULTI-LABEL CLASSIFICATION METHODS

- **CT.** For each multi-label training sample associated with m labels, the CT method creates m single-label samples, each of which can be associated with one label at a time [43]. Then a single-label learning method can be used to predict the label-set of the test instance.
- **BR.** The BR [38] method decomposes a multi-label classification task into several binary classification tasks (one-vs-rest). The BR technique assumes that all labels are independent, and each classifier is independently learned for each label prediction.
- **ECC.** The ECC method [44] links L single-label classifiers along a chain, and the inputs of each classifier is extended with the result of the proceeding classifiers in the chain.
- **DAG.** Lee *et al.* [45] report than instead of focusing on avoiding bad chain orders, finding an optimal classifiers' order in a chain help to improve the prediction accuracy. They build a DAG of labels using K2 algorithm with correlated ancestor set strategy [45] such that the correlations between parent and child nodes can be maximized. Specifically, they compute correlation with conditionally entropy and construct a DAG the maximizes the sum of conditional entropies between all parent-child nodes. Consequently, highly correlated labels are order in a chain obtained from DAG [45].

TABLE 1. Statistics of the corpora.

Corpora	Number of Authors	Number of Documents
Synthetic	2,720	7,200
Real Corpus (Computer Science)	707	1,957
Real Corpus (Social Sciences)	300	616
Real Corpus (Bioinformatics)	602	1,803

- **LSDM.** The LSDM method performs multi-label classification by exploring the most discrimintaive features associated with each class label. At first the LSDM method performs cluster analysis on positive and negative instances of the training data for each class label, and then reconstructs the feature spaces based on distance mapping and linear representation, by querying the clustering results for each class label. After that, the LSDM method employs sLDA (simplified linear discriminent analysis) to excavate the best feature space from reconstructed feature spaces of the identical class labels. Finally, the classifiers are learned using the excavated results [46].
- **MLkNN.** MLkNN is a popular multi-label classifier [51]. Similar to the regular k nearest neighbor classifier, MLkNN identifies the k closest neighbors corresponding to a test sample. To make a *multi-label prediction*, this classifier derives statistical information from the label set of the identified k closest neighbors; that is, the number of neighbors associated with each label. As a final step, MLkNN uses the principle of *Maximum A Posteriori (MAP)* to determine the label set of the test sample [51].

The parameter settings of each competitive method are as suggested in the corresponding literatures. We have used 5-fold cross-validation in order to evaluate each method.

B. CORPORA

we evaluate each method using one synthetic corpus and three real-world corpora. A detailed description of each corpus is as follows.

1) SYNTHETIC CORPUS

The evaluation of an AIMD solution requires a synthetic corpus of multi-author documents [12]. To generate a synthetic corpus, we obtain a collection of 23,096 *single-author* documents from the publicly available *Project Gutenberg*.[6] These documents were written by a set of 8,698 authors. From this set of authors, we select a subset A of authors who had written 10 or more documents of at least 12,000 tokens each unless stated otherwise. With the subset A of authors, we generate the *synthetic multi-author documents* as follows. Assume that a multi-author document D_a was written by m authors from A. We generate D_a by randomly choosing m authors $\{a_1, \ldots, a_m\}$ from A. For each author a_i, we obtain a text sample of length $\mathcal{L}/m$ tokens, where $\mathcal{L}$ denotes the length of the synthetic document. To avoid *training-testing* sample *contamination*, once a document is used to generate a synthetic document, it is never used again to generate another synthetic document. Moreover, each co-author set $\{a_1, \ldots, a_m\}$ of a multi-author document is unique. The statistics of the synthetic corpus are given in Table 1. By following the stated process, we obtain a synthetic corpus containing 7,200 multi-author documents from 2,720 authors. This synthetic corpus is 100% larger than the synthetic corpus used in the preliminary conference version of this work.

[6] https://www.gutenberg.org

TABLE 2. Default parameter values.

m	ϕ	$\mathcal{L}$	l	f	ω	θ	k	MHD
3	2	12,000	500	6,000	50	2,000	10	(50, 100]

2) REAL-WORLD CORPORA

To conduct experiments on real corpora, we download three collections of research articles: (i) social science from `scirp.org`; (ii) computer science from `arXiv.org` and (iii) bioinformatics from `biomedcentral.com`. We chose papers from these three collections such that each author has contributed to five or more research papers. The statistics of our real-world corpora are given in Table 1. For the *computer science* research papers, we obtain a set of 1,957 research papers written by 707 authors. For the *social science* and *bioinformatics* research papers, we obtained two sets of research papers with 616 and 1,803 papers written by 300 and 602 authors, respectively.

C. EXPERIMENT METHODOLOGY

In this section, we describe the *evaluation measures* and *parameter settings*.

TABLE 3. Synthetic corpus results: Effect of the number *m* of co-authors.

		Number m of Co-Authors							
		2		3		4		5	
	Method	Accuracy	Guess-one	Accuracy	Guess-one	Accuracy	Guess-one	Accuracy	Guess-one
Our Method	**CAG**	**94.52**	**97.99**	**93.79**	**97.86**	**92.88**	**97.69**	**93.10**	**97.03**
Existing AIMD Methods	B-CAG	73.39	86.25	61.96	85.19	59.87	85.22	58.27	85.13
	I-AICD	42.83	54.24	20.89	42.91	13.06	33.76	09.65	27.56
	AICD	35.67	44.26	09.38	29.77	06.99	26.68	05.07	19.68
Multi-label Classification Methods	BR	39.51	53.47	20.45	38.77	14.74	32.17	12.34	26.53
	CT	44.81	57.29	30.12	45.83	19.67	37.68	16.19	28.77
	ECC	46.48	59.79	32.02	48.85	23.96	41.78	16.81	28.88
	DAG	48.09	61.87	34.16	51.49	26.74	44.61	18.07	29.77
	MLkNN	48.15	63.87	33.47	52.09	27.98	43.55	20.18	32.21
	LSDM	47.37	64.46	41.82	57.72	28.69	46.53	23.46	37.96

1) PARAMETER SETTINGS

To identify the most appropriate value of each parameter, we test different values for each parameter. The following values have the best performance. For brevity, we present only the final parameter values used in this investigation unless stated otherwise. For the synthetic corpus, we fix the length of each synthetic multi-author document ($\mathcal{L}$) to 12,000 tokens. We fix the chunk size (l) and fragment size (f) to 500 tokens and 6 data points, respectively. For the chunk-level sliding window (ω) and fragment level sliding window (θ), increments of 50 tokens and 2 data points, respectively, have the best accuracy. The k value of 10 for top-k retrieval resulted in the best accuracy. For the *modified Hausdorff distance (MHD)* computation, the average of the ranked minimum distances falling into the (50%, 100%] range results in the best accuracy.

2) EVALUATION MEASURES

We use the following evaluation measures in the experiments.

i **Accuracy (A):** "Accuracy indicates the discrepancy of a prediction with respect to the ground truth, and is defined as the number of correctly predicted authors divided by the size of the true co-author set."

ii **Guess-one (G):** "A document is considered correct if the prediction contains at least one of the true authors."

D. PERFORMANCE EVALUATION

This section reports the findings obtained from our extensive experimental studies of the synthetic corpus. Recall that the main limitations of previous AIMD solutions are as follows: (i) the best existing stylometry-based AIMD solution has low accuracy, (ii) existing AIMD solutions report that increasing the number of co-authors of a paper adversely affects their performance and (iii) existing AIMD solutions cannot handle *non-writing authors (NWAs)*. Our experimental studies are designed such that we can verify whether our framework is capable of (i) effectively handling a larger number of co-authors, (ii) handling non-writing authors (NWAs) and (iii) handling multi-author documents of any length. Finally, we evaluate the proposed framework for four languages. Note that to control the (i) number m of co-authors, (ii) number ϕ of NWAs and (iii) number L of tokens for each multi-author document, these studies are performed using synthetic datasets. In addition, to show that the proposed solution can handle real-world datasets, we conduct experiments on three real-world corpora. The experimental results from these studies are reported in the following subsections.

1) PERFORMANCE EVALUATION WITH SYNTHETIC CORPUS

a: EFFECT OF THE NUMBER m OF AUTHORS

To study the effect of the number m of co-authors on the performance of each method, we vary the number m of

TABLE 4. Synthetic corpus results: Effect of the number ϕ of non-writing authors (NWAs).

		Number ϕ of Non-Writing Authors (NWAs)					
		0		1		2	
	Method	Accuracy	Guess-one	Accuracy	Guess-one	Accuracy	Guess-one
Our Method	**CAG**	**95.13**	**98.91**	**94.23**	**98.21**	**93.79**	**97.86**
Existing AIMD Methods	B-CAG	75.08	88.52	63.22	86.03	61.96	85.19
	I-AICD	47.82	55.36	29.57	49.14	20.89	42.91
	AICD	37.11	46.21	19.24	35.61	09.38	29.77
Multi-label Classification Methods	BR	46.75	55.12	29.11	47.67	20.45	38.77
	CT	47.08	58.87	32.44	48.37	30.12	45.83
	ECC	49.91	60.49	39.73	53.38	32.02	48.85
	DAG	52.11	65.85	41.03	59.12	34.16	51.49
	MLkNN	51.17	63.24	43.94	59.67	33.47	52.09
	LSDM	54.87	65.91	42.98	61.06	41.82	57.72

co-authors between 2 and 5. There are two motivations for doing so: (i) these values conform with the number m of co-authors in the real-word corpora used in this investigation and (ii) previous bibliometric analysis studies of the collaboration patterns of different fields report that the average number of co-authors per publication are 5 or fewer [52]–[55]. The experimental results obtained by varying the number m of co-authors are given in Table 3. As can be seen, our proposed method (CAG) outperforms all of the competitive techniques. In addition, increasing the number m of co-authors increases the performance gap between the proposed technique and the competitive techniques. Thus, the proposed solution can effectively handle larger numbers of co-authors with a higher accuracy than these competitive techniques. In terms of *guess-one* accuracy, the proposed method outperforms all of the competitive techniques.

b: EFFECT OF THE NUMBER ϕ OF NON-WRITING AUTHORS

To investigate the effect of varying the number ϕ of non-writing authors (NWAs) on the performance of each method, we vary the number ϕ between 0 and 2. The experimental results are shown in Table 4. Including NWAs in the actual list of co-authors marginally reduces the accuracy of the proposed framework in comparison to the competitive techniques. Because the competitive methods (except B-CAG) were not designed to handle non-writing co-authors, the accuracy of these methods decreases drastically as we increase the number ϕ of NWAs from 0 to 2. In terms of guess-one accuracy, the experimental results show that the proposed method is the best performer in all cases.

c: EFFECT OF THE NUMBER $\mathcal{L}$ OF DOCUMENT SIZE

In this study, we investigate the effect of varying the number $\mathcal{L}$ of document size on the performance of each method. As mentioned in the Introduction, our previously proposed framework can handle multi-author documents of 12,000 tokens or higher. In this research, we propose a solution that can also handle short publications. To study the effect of document size $\mathcal{L}$ on the performance of each method, we vary the document length $\mathcal{L}$ between 6,000 and 12,000 tokens.

The experimental results are given in Table 5. As can be seen, the proposed method can effectively handle documents of different lengths ranging between 6,000 to 12,000 tokens with better than 90% accuracy. In addition, our solution significantly outperforms the baseline method (B-CAG) and the competitors.

d: EFFECT OF THE NUMBER l OF CHUNK SIZE

We investigate the effect of chunk size on the performance of CAG, B-CAG, and I-AICD methods only, because the other methods are not designed to handle the chunk

TABLE 5. Synthetic corpus results: Effect of the number $\mathcal{L}$ of document size.

		Number $\mathcal{L}$ of Document Size							
		6,000		8,000		10,000		12,000	
	Method	Accuracy	Guess-one	Accuracy	Guess-one	Accuracy	Guess-one	Accuracy	Guess-one
Our Method	**CAG**	**90.87**	**94.23**	**91.55**	**95.89**	**92.11**	**93.79**	**93.79**	**97.86**
Existing AIMD Methods	B-CAG	56.17	77.29	56.29	80.93	56.97	81.37	61.96	85.19
	I-AICD	16.34	39.03	16.13	39.09	17.62	39.79	20.89	42.91
	AICD	07.56	22.78	07.62	23.61	08.04	25.19	09.38	29.77
Multi-label Classification Methods	BR	13.66	33.59	15.73	33.79	16.38	37.55	20.45	38.77
	CT	25.47	34.21	28.27	39.84	29.04	40.28	30.12	45.83
	ECC	28.37	39.54	29.76	42.84	29.87	43.77	32.02	48.85
	DAG	29.97	41.73	31.56	42.91	31.79	45.28	34.16	51.49
	MLkNN	31.09	44.38	31.93	45.19	32.67	79.21	33.47	52.09
	LSDM	37.19	49.26	37.82	51.33	41.35	52.47	41.82	57.72

TABLE 6. Synthetic corpus results: Effect of the number l chunk size.

		Number l of Chunk Size							
		500		750		1000		1250	
	Method	Accuracy	Guess-one	Accuracy	Guess-one	Accuracy	Guess-one	Accuracy	Guess-one
Our Method	**CAG**	**93.79**	**97.86**	**94.05**	**98.71**	**94.94**	**98.67**	**94.87**	**98.59**
Existing AIMD Methods	B-CAG	61.96	85.19	67.93	87.04	74.19	88.13	74.28	88.19
	I-AICD	20.89	42.91	20.16	48.09	26.14	48.93	27.34	49.03

size. We vary the chunk size between 500 and 1,250. The experimental results are shown in Table 6. This study has two main findings: (i) the proposed solution significantly outperforms all competitive techniques and (ii) for the proposed solution, a chunk size of 500 tokens marginally reduces accuracy compared to a chunk size of 1,250 tokens.

e: MULTILINGUAL AIMD

In this section, we present some experimental results in multi-lingual settings. To perform this experiment, we generate four synthetic corpora using the method given in Section IV-B, where each corpus is written in a different language: English, French, Finnish and German. We obtain 400 documents written by 100 authors for each cor-

TABLE 7. Multilingual synthetic dataset results.

Language	Accuracy	Guess-one
German	94.23	98.02
English	92.81	96.56
French	94.82	98.02
Finnish	94.23	97.76

TABLE 8. Real-world corpora results.

		Computer Science		Social Science		Bioinformatics	
	Method	Accuracy	Guess-one	Accuracy	Guess-one	Accuracy	Guess-one
Our Method	**CAG**	**75.34**	**99.43**	**51.69**	**99.12**	**77.42**	**99.61**
Existing AIMD Methods	B-CAG	54.13	95.37	40.98	91.54	58.03	96.12
	I-AICD	22.27	45.57	21.62	41.76	23.47	49.67
	AICD	13.25	30.43	19.67	36.54	16.12	33.57
Multi-label Classification Methods	BR	20.37	46.87	21.09	45.69	26.53	51.45
	CT	25.32	49.08	23.17	48.83	28.43	54.08
	ECC	26.49	54.22	26.07	54.88	25.98	53.29
	DAG	25.67	55.19	27.19	53.11	27.34	56.17
	MLkNN	29.18	57.09	28.49	58.07	29.97	59.31
	LSDM	28.39	57.49	26.97	55.16	29.74	59.86

pus. We fix the number of documents and number of authors in each corpus to fairly compare the performance for the different languages. The feature set contains 13 vocabulary richness-based features (feature 1 to feature 13 in Table 9), all of the features listed in the structural features in Table 9 and 12 *part-of-speech (POS)*-based features extracted using a universal part-of-speech tagger [56].

Note that in monolingual AIMD studies, the POS can be calculated using the best POS tagger available for a particular language. For example, for English, *Penn Treebank* classifies words into 36 linguistic categories [57]. Similarly, for French and German, *French Treebank* and *Stuttgart/Tübinger* classify words into 30 and 55 linguistic categories, respectively [57], [58]. However, when designing a multilingual AIMD solution, we need to take into account that the different granularities of the *linguistics categories* for different languages—in this case, 36, 30 and 55 *linguistic categories* for *English*, *French* and *German*, respectively—imply that they are not directly comparable. These different linguistic categories can be converted into a common set of linguistic categories for all languages to make the experimental results comparable across different languages. To perform such a multilingual research, we use the *universal POS tagger* [56]. The *universal POS tagger* categorizes words into 12 linguistic categories[7] that are universal across different languages in our corpora.

[7]Verb (verbs), Noun (nouns), Adv (adverbs), Adj (adjectives), Det (articles and determiners), Pron (pronouns), Num (numerals), Adp (prepositions and postpositions), Prt (particles), ”.” (punctuation marks), Conj (conjunctions) and X (all other categories, such as punctuation, foreign words or abbreviations.)

TABLE 9. List of stylometric features.

Lexical Features			
1. N: Total #words	2. V: Total #distinct words	3. Average word length	4. S.D. of word lengths
5. $\frac{V}{N}$	6. $VR(K) = \frac{10^4(\sum i^2 V_i - N)}{N^2}$	7. $VR(R) = \frac{V}{\sqrt{N}}$	8. $VR(C) = \frac{\log V}{\log N}$
9. $VR(H) = \frac{(100 \log N)}{(1-V_1)/V}$	10. $VR(S) = \frac{V_2}{V}$	11. $VR(k) = \frac{logV}{\log(\log N)}$	12. $VR(LN) = \frac{(1-V^2)}{V^2(\log N)}$
13. Entropy of word freq. ditri.	14. Total number of chars	15. Freq. of alpha chars	16. Freq. of uppercase chars
17. Freq. of lowercase chars	18. Freq. of numeric chars	19. Freq. of special chars	20. Freq. of white spaces
21. Freq. of punctuations	22. Alpha char ratio	23. Uppercase char ratio	24. Lowercase char ration
25. Numeric char ratio	26. Special char ratio	27. White spaces ratio	
Syntactic Features			
28. Freq. of nouns	29. Freq. of proper nouns	30. Freq. of pronouns	31. Freq. of ordinal adjs.
32. Freq. of comparative adjs.	33. Freq. of superlative adjs.	34. Freq. of advs.	35. Freq. of comparative advs.
36. Freq. of superlative advbs.	37. Freq. of modal auxiliaries	38. Freq. of bases form verbs	39. Freq. of past verbs
40. Freq. of present part. verbs	41. Freq. of past part. verbs	42. Freq. of particles	43. Freq. of wh-words
44. Freq. of conjunctions	45. Freq. of numerical words	46. Freq. of determiners	47. Freq. of existential theres
48. Freq. of existential to	49. Freq. of prepositions	50. Freq. of genitive markers	51. Freq. of quotations
52. Freq. of commas	53. Freq. of terminators	54. Freq. of symbols	
Structural Features			
55. Total number of sentence	56. Avg. #words per sentence	57-1056. Character n-gram	

Our feature space for multilingual AIMD relies on a minimal linguistic assumption set that includes (i) the ability to tokenize a text sample into words, (ii) the ability to identify sentence boundaries, (iii) the capability of POS tagging and (iv) the use of punctuation. The experimental results are given in Table 7. Because the competitive techniques were designed for English, we provide the results for the proposed technique only. The experimental results show that our proposed framework achieves better than 92% accuracy for each language.

2) PERFORMANCE EVALUATION WITH REAL-WORLD CORPORA

We also evaluate our proposed framework using three real-world corpora. Unlike the synthetic corpus of multi-author documents, where we have the ground truth information regarding the NWAs of each synthetic document, real-world corpora do not contain ground truth information. Hence, while measuring accuracy for real-world corpora, we assume that all of the listed authors on a paper contributed to writing the paper. This assumption reduces the measured accuracy values of all techniques in comparison to their actual values. However, it allows us to compare all of the methods for real-world corpora. The experimental results are given in Table 8. We can see that our proposed solution significantly outperforms the other methods. Because ground truth information on the NWAs is not available in real-world corpora, the accuracy of the proposed technique in this study is lower than that for the synthetic corpus (see Section IV-D.1).

V. CONCLUSION

In this study, we propose an effective and scalable framework for performing authorship identification on multi-author documents. The primary contribution of our proposed framework lies in its capability to probabilistically attribute different fragments (parts) of the same multi-author document to different authors on its author list. Specifically, our proposed *Co-Authorship* Graph (CAG) data structure

can capture stylistic similarities between pairs of fragments across the entire document corpus. In addition, our graph training algorithm is effective at (i) learning the true writer(s) of each fragment and (ii) identifying the NWAs of multi-author documents. Further, along with our previous feature space, the character *n*-gram-based features have proven to perform well in solving AIMD problem regardless of the length of text samples. Moreover, extracting character *n*-grams does not require tokenizers, taggers, parsers or any language-dependent and non-trivial NLP tools, which makes them feasible for performing *multi-lingual* authorship attribution tasks. We evaluate our framework and competitive techniques on one synthetic corpus and three real-world corpora. Our extensive experimental studies show that our proposed framework (i) significantly outperforms competitive techniques, (ii) can more effectively handle a larger number of co-authors than competitive techniques and (iii) can effectively handle NWAs in multi-author documents.

VI. APPENDIX
STYLOMETRIC FEATURES

The stylometric features used in this investigation are shown in Table 9. For features 5 to 12, N represents the count of words and V represents the count of distinct words. For Features 6 and 9, V_i represents the count of words that occur i times. For the multi-lingual experiments, the part-of-speech based features are projected to the *universal part-of-speech tag set*. Note that character *n*-grams are categorized as lexical features. However, in Table 9, we list them under the category of structural features to realize the concept of feature concatenation used in this work.

REFERENCES

[1] R. Sarwar, T. Porthaveepong, A. Rutherford, T. Rakthanmanon, and S. Nutanong, "Stylothai: A scalable framework for stylometric authorship identification of thai documents," *ACM Trans. Asian Low-Resource Lang. Inf. Process. (TALLIP)*, vol. 19, no. 3, pp. 1–15, 2020.

[2] S. Nutanong, C. Yu, R. Sarwar, P. Xu, and D. Chow, "A scalable framework for stylometric analysis query processing," in *Proc. IEEE 16th Int. Conf. Data Mining (ICDM)*, Dec. 2016, pp. 1125–1130.

[3] R. Sarwar, Q. Li, T. Rakthanmanon, and S. Nutanong, "A scalable framework for cross-lingual authorship identification," *Inf. Sci.*, vol. 465, pp. 323–339, Oct. 2018.

[4] R. Sarwar, C. Yu, N. Tungare, K. Chitavisutthivong, S. Sriratanawilai, Y. Xu, D. Chow, T. Rakthanmanon, and S. Nutanong, "An effective and scalable framework for authorship attribution query processing," *IEEE Access*, vol. 6, pp. 50030–50048, 2018.

[5] S. H. H. Ding, B. C. M. Fung, and M. Debbabi, "A visualizable evidence-driven approach for authorship attribution," *ACM Trans. Inf. Syst. Secur.*, vol. 17, no. 3, pp. 1–30, Mar. 2015.

[6] J. Savoy, "Authorship attribution based on specific vocabulary," *ACM Trans. Inf. Syst.*, vol. 30, no. 2, pp. 1–30, May 2012.

[7] R. Sarwar and S. Nutanong, "The key factors and their influence in authorship attribution," *Res. Comput. Sci.*, vol. 110, pp. 139–150, Feb. 2016.

[8] A. Rocha, W. J. Scheirer, C. W. Forstall, T. Cavalcante, A. Theophilo, B. Shen, A. R. B. Carvalho, and E. Stamatatos, "Authorship attribution for social media forensics," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 1, pp. 5–33, Jan. 2017.

[9] E. Dauber, R. Overdorf, and R. Greenstadt, "Stylometric authorship attribution of collaborative documents," in *Proc. Int. Conf. Cyber Secur. Cryptogr. Mach. Learn. (CSCML)*, Beer-Sheva, Israel, Jun. 2017, pp. 115–135.

[10] H. V. Agun and O. Yilmazel, "Incorporating topic information in a global feature selection schema for authorship attribution," *IEEE Access*, vol. 7, pp. 98522–98529, 2019.

[11] W. Anwar, I. S. Bajwa, M. A. Choudhary, and S. Ramzan, "An empirical study on forensic analysis of urdu text using LDA-based authorship attribution," *IEEE Access*, vol. 7, pp. 3224–3234, 2019.

[12] R. Sarwar, C. Yu, S. Nutanong, N. Urailertprasert, N. Vannaboot, and T. Rakthanmanon, "A scalable framework for stylometric analysis of multi-author documents," in *Proc. Int. Conf. Database Syst. Adv. Appl. (DASFAA)*, Gold Coast, QLD, Australia, May 2018, pp. 813–829.

[13] R. Zhang, Z. Hu, H. Guo, and Y. Mao, "Syntax encoding with application in authorship attribution," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Oct./Nov. 2018, pp. 2742–2753.

[14] J. K. Bradley, P. G. Kelley, and A. Roth, "Author identification from citations," Dept. Comput. Sci., Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep, 2008.

[15] S. Hill and F. Provost, "The myth of the double-blind review?: Author identification using only citations," *ACM SIGKDD Explor. Newslett.*, vol. 5, no. 2, pp. 179–184, 2003.

[16] M. Payer, L. Huang, N. Z. Gong, K. Borgolte, and M. Frank, "What you submit is who you are: A multimodal approach for deanonymizing scientific publications," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 1, pp. 200–212, Jan. 2015.

[17] E. Stamatatos, "On the robustness of authorship attribution based on character n-Gram features," *J. Law Policy*, vol. 21, no. 2, pp. 421–439, 2013.

[18] V. Kešelj, F. Peng, N. Cercone, and C. Thomas, "N-gram-based author profiles for authorship attribution," in *Proc. Conf. Pacific Assoc. Comput. Linguistics (PACLING)*, vol. 3, 2003, pp. 255–264.

[19] E. Stamatatos, "Ensemble-based author identification using character n-grams," in *Proc. 3rd Int. Workshop Text-Based Inf. Retr.*, vol. 36, 2006, pp. 41–46.

[20] E. Stamatatos, "A survey of modern authorship attribution methods," *J. Amer. Soc. Inf. Sci.*, vol. 60, no. 3, pp. 538–556, Mar. 2009.

[21] F. Peng, D. Schuurmans, and S. Wang, "Augmenting naive Bayes classifiers with statistical language models," *Inf. Retr.*, vol. 7, nos. 3–4, pp. 317–345, Sep. 2004.

[22] F. Peng, D. Schuurmans, S. Wang, and V. Keselj, "Language independent authorship attribution using character level language models," in *Proc. 10th Conf. Eur. Chapter Assoc. Comput. Linguistics*, vol. 1, 2003, pp. 267–274.

[23] J. Peng, K. R. Choo, and H. Ashman, "Astroturfing detection in social media: Using binary n-Gram analysis for authorship attribution," in *Proc. IEEE Trustcom/BigDataSE/ISPA*, Tianjin, China, Aug. 2016, pp. 121–128.

[24] I. Kanaris, K. Kanaris, I. Houvardas, and E. Stamatatos, "Words versus character n-grams for anti-spam filtering," *Int. J. Artif. Intell. Tools*, vol. 16, no. 06, pp. 1047–1067, Dec. 2007.

[25] J. Sun, Z. Yang, P. Wang, and S. Liu, "Variable length character n-Gram approach for online writeprint identification," in *Proc. Int. Conf. Multimedia Inf. Netw. Secur.*, 2010, pp. 486–490.

[26] L. Muttenthaler, G. Lucas, and J. Amann, "Authorship attribution in fan-fictional texts given variable length character and word n-grams," in *Proc. Work. Notes Conf. Labs Eval. Forum (CLEF)*, Lugano, Switzerland, Sep. 2019.

[27] T. Amano, J. P. González-Varo, and W. J. Sutherland, "Languages are still a major barrier to global science," *PLoS Biol.*, vol. 14, no. 12, Jan. 2017, Art. no. e2000933.

[28] S. H. H. Ding, B. C. M. Fung, F. Iqbal, and W. K. Cheung, "Learning stylometric representations for authorship analysis," *IEEE Trans. Cybern.*, vol. 49, no. 1, pp. 107–121, Jan. 2019.

[29] Z. Jiang, S. Yu, Q. Qu, M. Yang, J. Luo, and J. Liu, "Multi-task learning for author profiling with hierarchical features," in *Proc. Companion Web Conf.*, Lyon, France, Apr. 2018, pp. 55–56

[30] O. De Vel, A. Anderson, M. Corney, and G. Mohay, "Mining e-mail content for author identification forensics," *SIGMOD Rec.*, vol. 30, no. 4, pp. 55–64, Dec. 2001.

[31] M. van Dam and C. Hauff, "Large-scale author verification: Temporal and topical influences," in *Proc. 37th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, Gold Coast, QLD, Australia, Jul. 2014, pp. 1039–1042.

[32] J. Grieve, "Quantitative authorship attribution: An evaluation of techniques," *Literary Linguistic Comput.*, vol. 22, no. 3, pp. 251–270, Jul. 2007.
[33] J. Li, R. Zheng, and H. Chen, "From fingerprint to writeprint," *Commun. ACM*, vol. 49, no. 4, pp. 76–82, Apr. 2006.
[34] M. G. Kendall, F. Mosteller, and D. L. Wallace, "Inference and disputed authorship: The Federalist," *Biometrics*, vol. 22, no. 1, p. 200, Mar. 1966.
[35] C. E. Chaski, "Empirical evaluations of language-based author identification techniques," *Forensic Linguistics*, vol. 8, no. 1, pp. 1–65, Jun. 2001.
[36] A. Abbasi and H. Chen, "Writeprints: A stylometric approach to identity-level identification and similarity detection in cyberspace," *ACM Trans. Inf. Syst.*, vol. 26, no. 2, pp. 1–29, Mar. 2008.
[37] S.-U. Hassan, R. Sarwar, and A. Muazzam, "Tapping into intra- and international collaborations of the Organization of Islamic Cooperation states across science and technology disciplines," *Sci. Public Policy*, vol. 43, no. 5, pp. 690–701, Oct. 2016.
[38] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown, "Learning multi-label scene classification," *Pattern Recognit.*, vol. 37, no. 9, pp. 1757–1771, Sep. 2004.
[39] S.-U. Hassan, N. R. Aljohani, N. Idrees, R. Sarwar, R. Nawaz, E. Martínez-Cámara, S. Ventura, and F. Herrera, "Predicting literature's early impact with sentiment analysis in Twitter," *Knowl.-Based Syst.*, to be published, doi: 10.1016/j.knosys.2019.105383.
[40] S. A. Alanazi, "Toward identifying features for automatic gender detection: A corpus creation and analysis," *IEEE Access*, vol. 7, pp. 111931–111943, 2019.
[41] J. E. Tapia and C. A. Perez, "Gender classification from NIR images by using quadrature encoding filters of the most relevant features," *IEEE Access*, vol. 7, pp. 29114–29127, 2019.
[42] A. Elisseeff and J. Weston, "A kernel method for multi-labelled classification," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 681–687.
[43] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," *Int. J. Data Warehousing Mining*, vol. 3, no. 3, pp. 1–13, 2007.
[44] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," *Mach. Learn.*, vol. 85, no. 3, pp. 333–359, Dec. 2011.
[45] J. Lee, H. Kim, N.-R. Kim, and J.-H. Lee, "An approach for multi-label classification by directed acyclic graph with label correlation maximization," *Inf. Sci.*, vol. 351, pp. 101–114, Jul. 2016.
[46] Y. Guo, F. Chung, G. Li, J. Wang, and J. C. Gee, "Leveraging label-specific discriminant mapping features for multi-label learning," *ACM Trans. Knowl. Discov. Data*, vol. 13, no. 2, pp. 1–23, Apr. 2019.
[47] R. Lipikorn, A. Shimizu, and H. Kobatake, "A modified Hausdorff distance for object matching," in *Proc. 12th Int. Conf. Pattern Recognit.*, vol. 1, 1994, pp. 566–568.
[48] D. Huttenlocher, G. Klanderman, and W. Rucklidge, "Comparing images using the Hausdorff distance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 9, pp. 850–863, Sep. 1993.
[49] C. C. Holmes and N. M. Adams, "A probabilistic nearest neighbour method for statistical pattern recognition," *J. Roy. Stat. Soc., Ser. B (Stat. Methodol.)*, vol. 64, no. 2, pp. 295–306, May 2002.
[50] A. W. E. McDonald, S. Afroz, A. Caliskan, A. Stolerman, and R. Greenstadt, "Use fewer instances of the letter 'i': Toward writing style anonymization," in *Proc. 12th Int. Symp. Privacy Enhancing Technol. Symp. (PETS)*, Vigo, Spain, Jul. 2012, pp. 299–318.
[51] M.-L. Zhang and Z.-H. Zhou, "ML-KNN: A lazy learning approach to multi-label learning," *Pattern Recognit.*, vol. 40, no. 7, pp. 2038–2048, Jul. 2007.
[52] P. Akhavan, N. A. Ebrahim, M. A. Fetrati, and A. Pezeshkan, "Major trends in knowledge management research: A bibliometric study," *Scientometrics*, vol. 107, no. 3, pp. 1249–1264, Jun. 2016.
[53] A. Geminiani, C. Ercoli, C. Feng, and J. G. Caton, "Bibliometrics study on authorship trends in periodontal literature from 1995 to 2010," *J. Periodontol.*, vol. 85, no. 5, pp. e136–e143, May 2014.
[54] R. Sarwar, S. H. Soroya, A. Muazzam, F. Sabah, S. Iqbal, and S.-U. Hassan, "A bibliometric perspective on technology-driven innovation in the gulf cooperation council (GCC) countries in relation to its transformative impact on international business," in *Technology-Driven Innovation in Gulf Cooperation Council (GCC) Countries: Emerging Research and Opportunities*. Hershey, PA, USA: IGI Global, 2019, pp. 49–66.
[55] R. Sarwar and S.-U. Hassan, "A bibliometric assessment of scientific productivity and international collaboration of the Islamic World in science and technology (S&T) areas," *Scientometrics*, vol. 105, no. 2, pp. 1059–1077, Nov. 2015.
[56] S. Petrov, D. Das, and R. McDonald, "A universal part-of-speech tagset," 2011, *arXiv:1104.2086*. [Online]. Available: https://arxiv.org/abs/1104.2086
[57] S. Malmasi and M. Dras, "Multilingual native language identification," *Nat. Lang. Eng.*, vol. 23, no. 2, pp. 163–215, Mar. 2017.
[58] A. Abeillé, L. Clément, and F. Toussenel, "Building a treebank for french," in *Proc. 2nd Int. Conf. Lang. Resour. Eval. (LREC)*, Athens, Greece. Paris, France: European Language Resources Association, May/Jun. 2000, pp. 165–187.

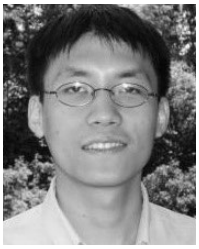
RAHEEM SARWAR received the Ph.D. degree in computer science from the City University of Hong Kong. He is a currently Postdoctoral Fellow with the School of Information Science and Technology, Vidyasirimedhi Institute of Science and Technology (VISTEC), Thailand. His research interests include stylometry, digital text forensics, data science, and large-scale machine learning.

NORAWIT URAILERTPRASERT received the bachelor's degree in computer engineering from the Department of Computer Engineering, Kasetsart University, Thailand. He is currently pursuing the Ph.D. degree with the School of Information Science and Technology, Vidyasirimedhi Institute of Science and Technology (VISTEC), Thailand. His research interests include computer vision, stylometry, deep learning, and text classification.

NATTAPOL VANNABOOT received the bachelor's degree in computer engineering from the Department of Computer Engineering, Kasetsart University, Thailand. He is currently a Research Assistant with the School of Information Science and Technology, Vidyasirimedhi Institute of Science and Technology (VISTEC), Thailand. His research interests include stylometry and scalable machine learning.

CHENYUN YU received the Ph.D. degree in computer science from the City University of Hong Kong. She is currently a Postdoctoral Fellow with the Department of Computer Science, National University of Singapore. Her research interests include data processing, query optimization, and large-scale machine learning.

THANAWIN RAKTHANMANON received the Ph.D. degree in computer science from the University of California, USA. He is currently an Assistant Professor with the Department of Computer Engineering, Kasetsart University, Thailand, and the School of Information Science and Technology, VISTEC. He has authored over 35 journals articles and conference papers. His research interests include time series mining, data mining, and large-scale machine learning. He has received the SIGKDD 2012 Best Paper Award and the Young Innovative Award in computer science applications from Office of the Higher Education Commission, Thailand.



EKAPOL CHUANGSUWANICH received the Ph.D. degree in electrical engineering and computer science from the Massachusetts Institute of Technology, USA. He is currently an Assistant Professor with the Computer Engineering Department, Faculty of Engineering, Chulalongkorn University. His research interests include automatic speech recognition and machine learning.

SARANA NUTANONG received the Ph.D. degree from the University of Melbourne, Australia.

He held an assistant professor position at the City University of Hong Kong. He is currently an Associate Professor with the School of Information Science and Technology, Vidyasirimedhi Institute of Science and Technology (VISTEC), Thailand. He has authored over 40 journal and conference publications. His research interests include scientific data management, data-intensive computing, spatial-temporal query processing, and large-scale machine learning. He has served as a PC Member and an Invited Reviewer for the IEEE TKDE, VLDB Journal, ACM SIGSPATIAL GIS, ADC, the IEEE TMC, and *Distributed and Parallel Databases*, in 2012.